

CLAIMS

1. (Currently amended) A method for executing a target application on a host processor comprising:

interpreting each of a sequence of target instructions to produce a sequence of host instructions and executing the interpreted target sequence of host instructions;

dynamically translating into host instructions each of a sequence of target instructions to produce translated host instructions, wherein the translating is performed when the number of times a sequence of target instructions is executed exceeds a preset count;

storing the translated host instructions wherein a plurality of additional registers are used during execution for holding the official state of a target processor;

responding to an exception during execution of a stored translated host instruction by rolling back to a previous point in execution at which correct state of a target processor is known; and

interpreting each target instruction in order from the point in execution at which a correct state of a target processor is known.

2. (Currently amended) The method of as claimed in Claim 1 which further comprises:

collecting statistics regarding the execution of sequences of instructions which are interpreted.

3. (Currently amended) A method for executing a target application on a host processor comprising the ~~steps of~~:

interpreting sequentially each of a sequence of ~~host-target~~ instructions to produce interpreted host instructions and executing the interpreted host instructions representing each target instruction of the target application;

performing a ~~dynamic~~-translation process when the number of times a host instruction is executed exceeds a preset count to produce translated host instructions wherein a plurality of additional registers are used during execution for holding the official state of a target processor;

responding to an exception during execution of the translated host instructions representing a target instruction by returning to a previous point in execution of the target application at which a correct state of a target processor is known; and

thereafter executing host instructions by interpretation of the target instruction until the point of the exception.

4. (Currently amended) ~~A-The~~ method as claimed in Claim 3 which further comprises collecting statistics regarding the execution of sequences of target instructions which are executed.

5. (Currently amended) ~~A-The~~ method as claimed in Claim 4 in which the statistics include the number of times the sequence of target instructions have executed.

6. (Currently amended) ~~A~~The method as claimed in Claim 4 in which the statistics include an address of an instruction to which a target instruction including a branch operation branches.

7. (Currently amended) ~~A~~The method as claimed in Claim 4 in which the statistics include a likelihood of a branch being taken.

8. (Currently amended) A computer ~~system-readable medium encoded with a computer program for implementing a system for executing a target application designed for execution on a target processor on a host processor having an instruction set different than that of the target processor comprising:~~

a bus;

a memory unit coupled with the bus, wherein said memory unit is operable to store computer-readable instructions for executing a target application designed for execution on a target processor on a host processor having an instruction set different than that of the target processor; and

a processor coupled with the bus, wherein when the computer readable instructions comprise:

means for instructions to interpreting a sequence of target instructions to produce a sequence of host instructions and executing each of the interpreted target sequences of host instructions;

means for instructions to dynamically translating translate sequences of target instructions to produce translated sequences of instructions and storing each translated

~~sequence of instructions, wherein the translating is performed~~ when the number of times a sequence of target instructions is executed exceeds a preset count;

~~means for instructions to storing~~ ~~store~~ each translated sequence of instructions wherein a plurality of additional registers are used during execution for holding the official state of the target processor,

~~means for instructions to responding~~ to an exception during execution of a stored translated instruction by rolling back to a previous point in execution at which correct state of a target processor is known, and

~~means for instructions to interpreting~~ each target instruction in order from a point in execution at which a correct state of a target processor is known through the target instruction causing the exception.

9. (Currently amended) ~~A-The~~ system as claimed in Claim 8 in which the ~~means for instructions to interpreting is comprise~~ an interpreter software executing on the host processor, and

the ~~means for instructions to translating~~ ~~translate is comprise~~ dynamic translation software executing on the host processor.

10. (Canceled)

11. (Canceled)

12. (Canceled)

13. (Canceled)

14. (Canceled)

15. (Currently amended) The method ~~of as claimed in~~ Claim 1, further comprising:

speculatively translating target instructions into host instructions based on a likelihood of a branch being taken.

16. (Currently amended) The method as claimed in of Claim 1, further comprising:

collecting statistics regarding the execution of sequences of target instructions which are executed.

17. (Currently amended) The method as claimed in of Claim 16, wherein the statistics include the number of times a branch to target instructions have executed.

18. (Currently amended) The method as claimed in of Claim 17, further comprising:

speculatively translating target instructions into host instructions based on a likelihood of a branch being taken.

19. (Currently amended) The ~~system~~ computer-readable medium ~~of as~~ claimed in Claim 8, further comprising:

instructions to collecting statistics including the number of times a branch of target instructions have executed.

20. (Currently amended) The system computer-readable medium of Claim 19, further comprising:

instructions to speculatively translating target instructions into host instructions based on a likelihood of a branch being taken.

21. (Currently amended) A method for executing a target application on a host processor comprising:

interpreting each of a sequence of target instructions to produce host instructions and executing the interpreted target host instructions;

dynamically translating into translated host instructions each of a sequence of target instructions, wherein the translating is performed when the number of times a sequence of target instructions is executed exceeds a preset count;

storing the translated host instructions wherein a plurality of additional registers are used during execution for holding the official state of a target processor and wherein a buffer stores working memory state changes and official memory state changes;

responding to an exception during execution of a stored translated host instruction by rolling back to a previous point in execution at which a correct state of a target processor is known; and

interpreting each target instruction in order from the point in execution at which the correct state of a target processor is known.

22. (Currently amended) A method for executing a target application on a host microprocessor, the method comprising:

interpreting each of a sequence of target operations to produce host operations and executing the ~~interpreted target~~ host operations;

~~dynamically~~ translating into host operations each of a sequence of target operations to produce translated host operations, wherein said translating is performed when the number of times a sequence of target operations is executed exceeds a predetermined count;

storing the translated host operations, wherein a plurality of additional registers are used during execution for holding the official state of a target microprocessor;

responding to an exception during execution of a stored translated host operation by rolling back to a previous time in execution at which a correct state of a target microprocessor is known; and

interpreting each target operation in order from the time in execution at which the correct state of a target microprocessor is known.

23. (Currently amended) The method as claimed in of Claim 22, further comprising:

collecting statistics regarding the execution of sequences of operations which are interpreted.

24. (Currently amended) The method as claimed in of Claim 22, further comprising:

speculatively translating target operations into host instructions based on a likelihood of a branch being taken.

25. (Currently amended) The method as claimed in of Claim 1, further comprising:

collecting statistics regarding the execution of sequences of target operations which are executed.

26. (Currently amended) The method as claimed in of Claim 25, wherein the statistics comprise the number of times a branch of target operations have executed.

27. (Currently amended) The method as claimed in of Claim 26, further comprising:

speculatively translating target operations into host operations based on a likelihood of a branch being taken.

28. (Currently Amended) ~~A computer readable medium having stored thereon computer executable instructions that, if executed by a system, cause the system to perform a method of executing instructions, the method comprising~~ A tangible computer-readable medium having instructions stored thereon, the instructions comprising:

instructions to interpreting a sequence of target instructions to produce host instructions and executing execute each of the interpreted target host instructions;
instructions to dynamically translating translate sequences of target instructions to produce sequences of host instructions and storing each translated sequence of instructions, wherein said translating is performed when the number of times a sequence of target instructions is executed exceeds a preset count;

instructions to storing store each translated sequence of instructions, wherein a plurality of additional registers are used during execution for holding the official state of the target processor;

instructions to responding to an exception during execution of a stored translated instruction by rolling back to a previous point in execution at which a correct state of a target processor is known; and

instructions to interpreting each target instruction in order from a point in execution at which the correct state of a target processor is known through the target instruction causing the exception.

29. (Currently amended) The computer-readable medium as claimed in ~~of~~ Claim 28, wherein said instructions to interpreting is performed by ~~comprise~~ an interpreter software executing on a host processor, and wherein said instructions to

~~translating~~ ~~translate is comprise~~ performed by dynamic translation software executing on the host processor.

30. (Currently amended) The computer-readable medium as claimed in of Claim 28, wherein the ~~method~~ instructions further comprises:

instructions to collecting statistics including the number of times a branch of target instructions have executed.

31. (Currently amended) The computer-readable medium as claimed in of Claim 30, wherein the ~~method~~ instructions further comprises:

instructions to speculatively translating ~~translate~~ target instructions into host instructions based on a likelihood of a branch being taken.